9/6/19 Discussion

Problem with model-based methods:

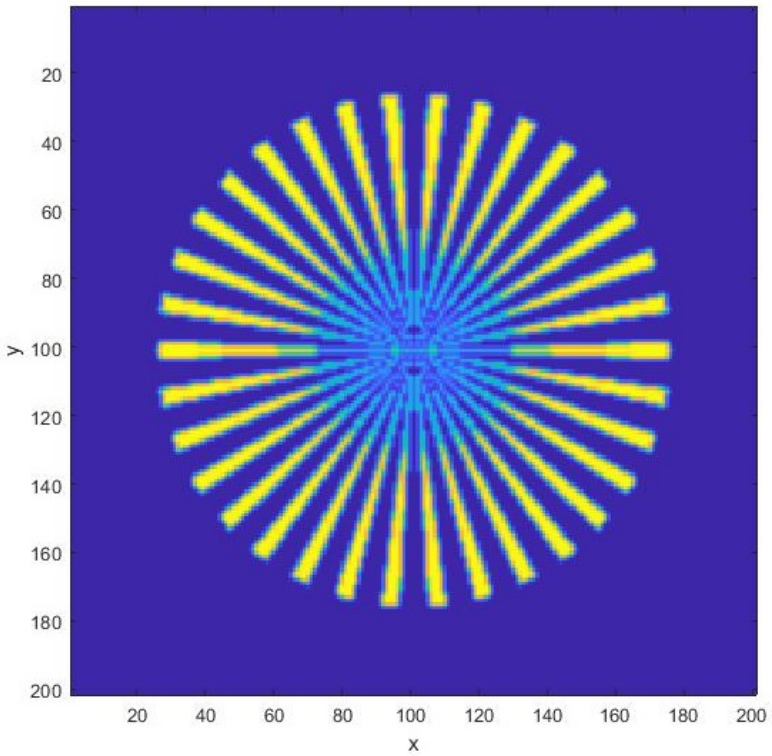
1. large memory requirement because of caching the up-sampled data g^{HR}, i_m, j_m and FFT/IFFT computation, for example, to compute $g(\boldsymbol{x}, z) = \sum_{m=1}^{N_m} [o(\boldsymbol{x}, z) j_m(\boldsymbol{x})] \circledast [h(\boldsymbol{x}, z) i_m(z)]$ in the forward model.
2. low performance due to the FFT/IFFT computation.

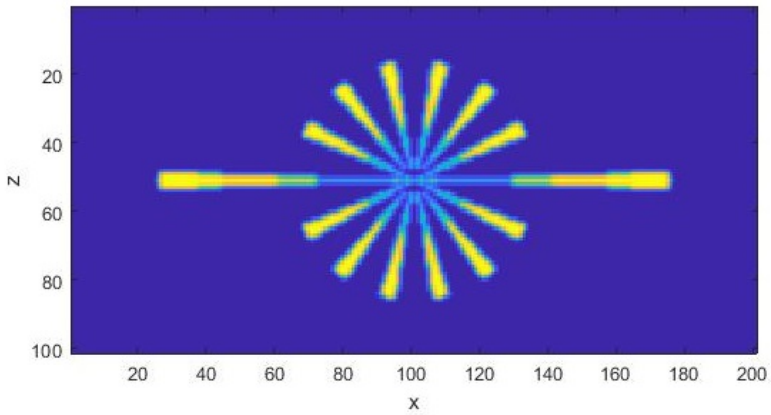
Improvement ideas using GPU: 3x faster performance and 4x less memory requirement when

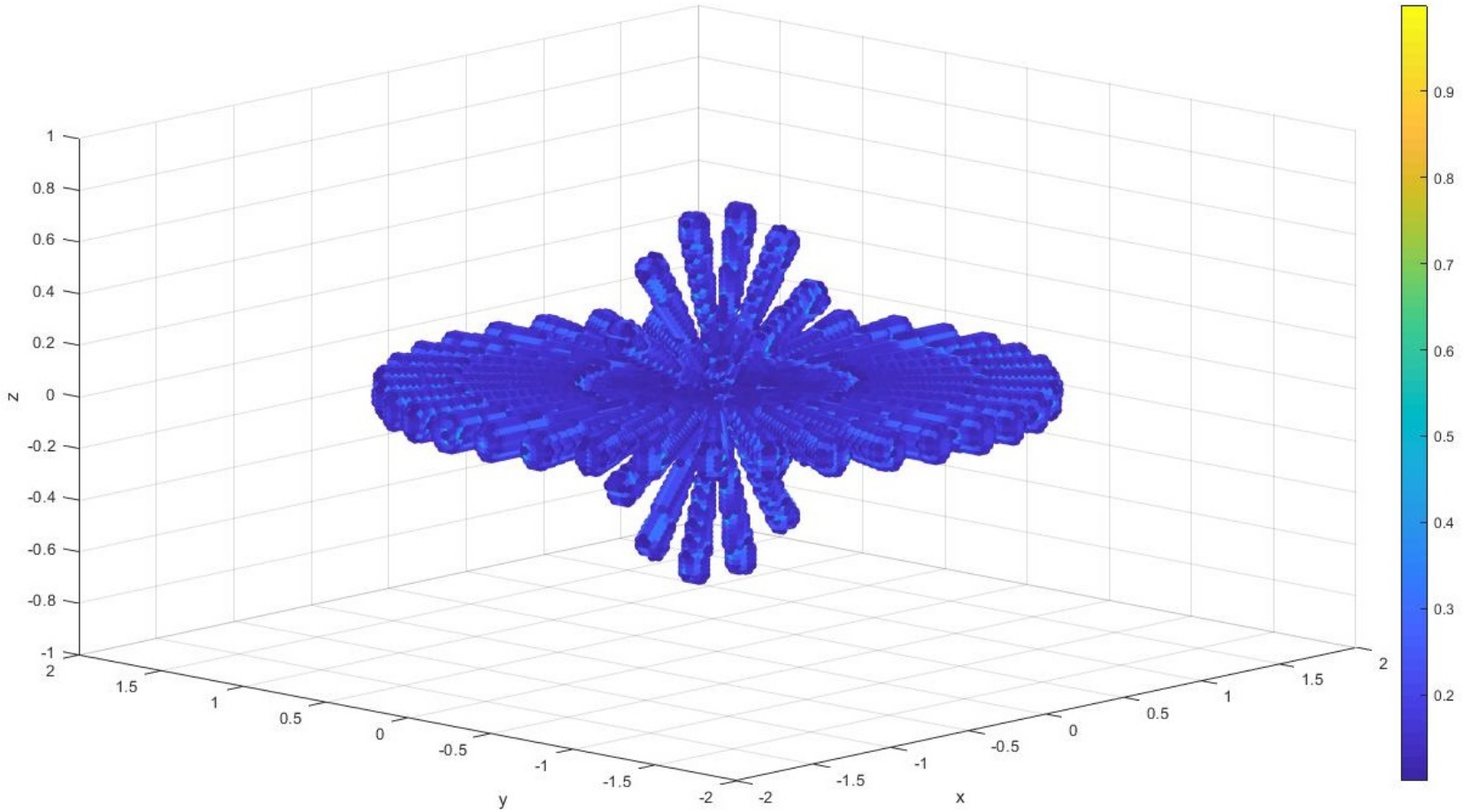
1. computing the up-sampled data, i_m, j_m on the fly using GPU.
2. doing the FFT/IFFT on GPU.

Problem still: doing FFT/IFFT on GPU requires large GPU memory. Maybe one only needs to use GPU to compute i_m, j_m and to up-sample the LR data (note that up-sampling the LR data requires FFT/IFFT).

On xy, xz planes






z
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
y
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
x
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2

1. FairSIM data: try model-based methods with FairSIM data, investigate parameters in FairSIM.
2. Run simulation with data reduction and model-based methods on current multi-object, for both 3W and Tunable.
3. Formulate 3D star-like object and run simulation on 3D star-like object.